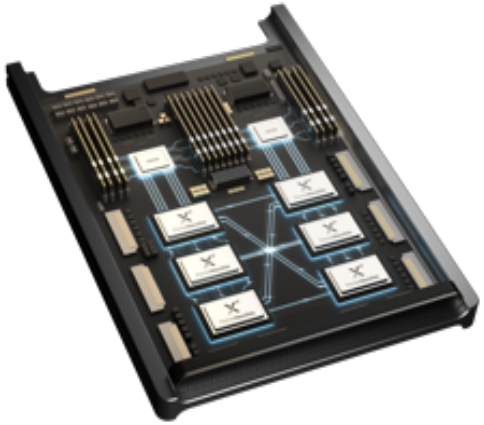
Aurora – Exascale at Argonne

P3HPC Forum Meeting, Sept 1-2
Scott Parker
Argonne Leadership Computing Facility

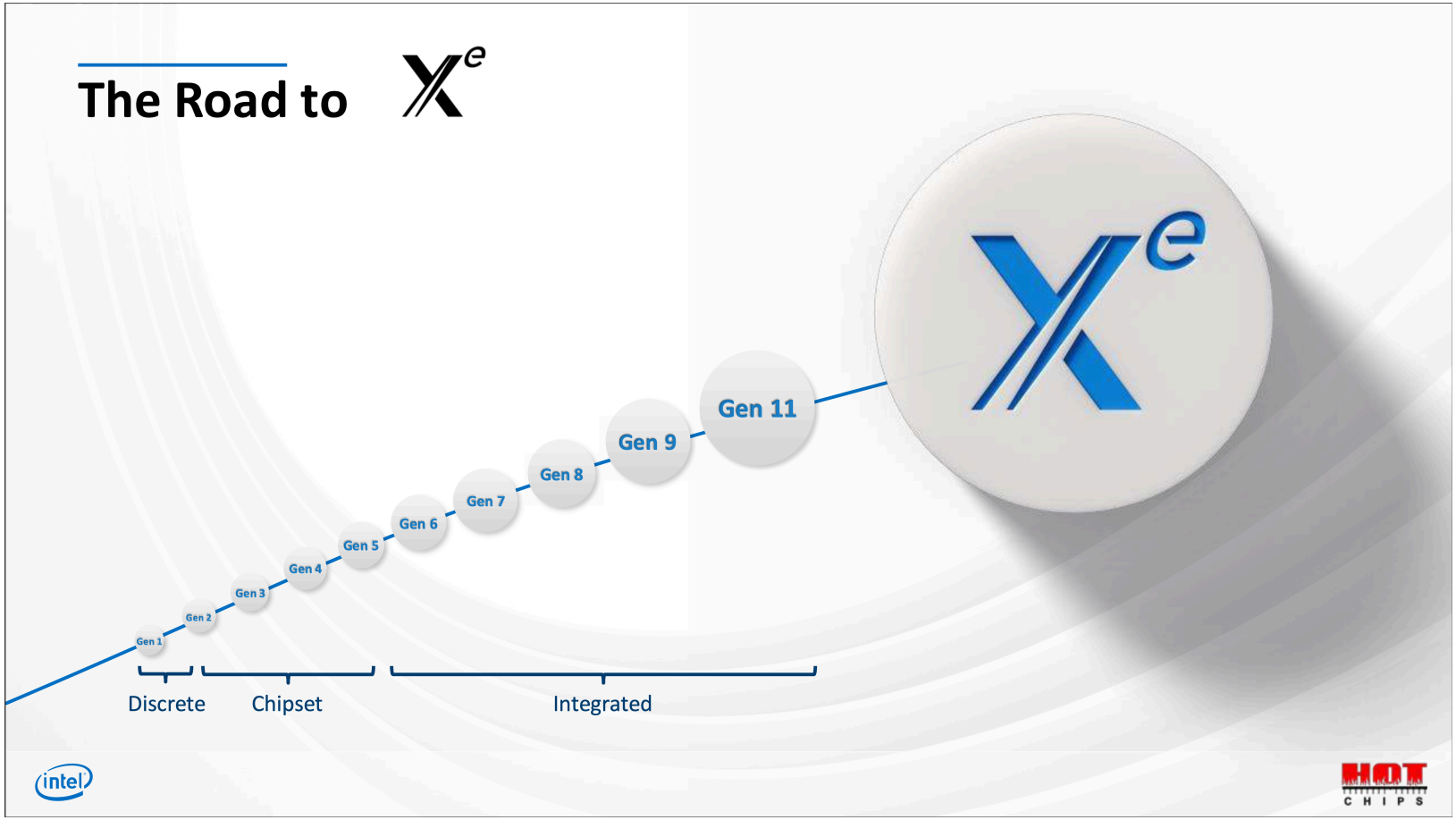
www.anl.gov

Aurora: A High-level View

- Intel-Cray machine arriving at Argonne in 2021:
  - Sustained Performance > 1 Exaflops
  - Greater than 10 PB of total memory
- Node has Intel Xeon processors and Intel X^e GPUs:
  - 2 Xeons (Sapphire Rapids)
  - 6 GPUs (Ponte Vecchio [PVC])
  - Unified Memory Architecture across CPUs and GPUs
- Cray Slingshot fabric and Shasta platform:
  - 8 endpoints per node
- Novel high-performance filesystem:
  - Distributed Asynchronous Object Store (DAOS)
    - ≥ 230 PB of storage capacity
    - Bandwidth of > 25 TB/s
  - Lustre
    - 150 PB of storage capacity
    - Bandwidth of ~1TB/s

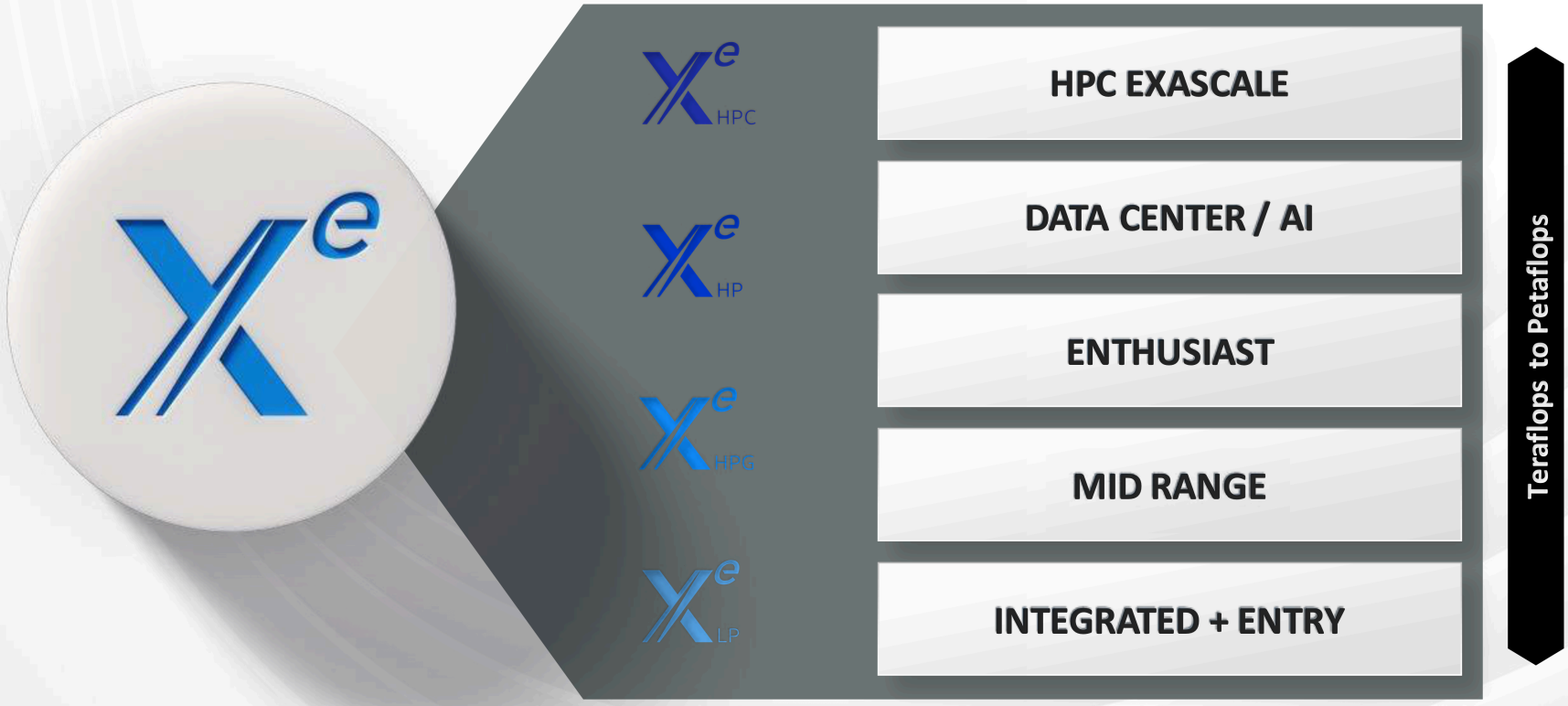


Evolution of Intel GPUs



Intel GPU Architecture

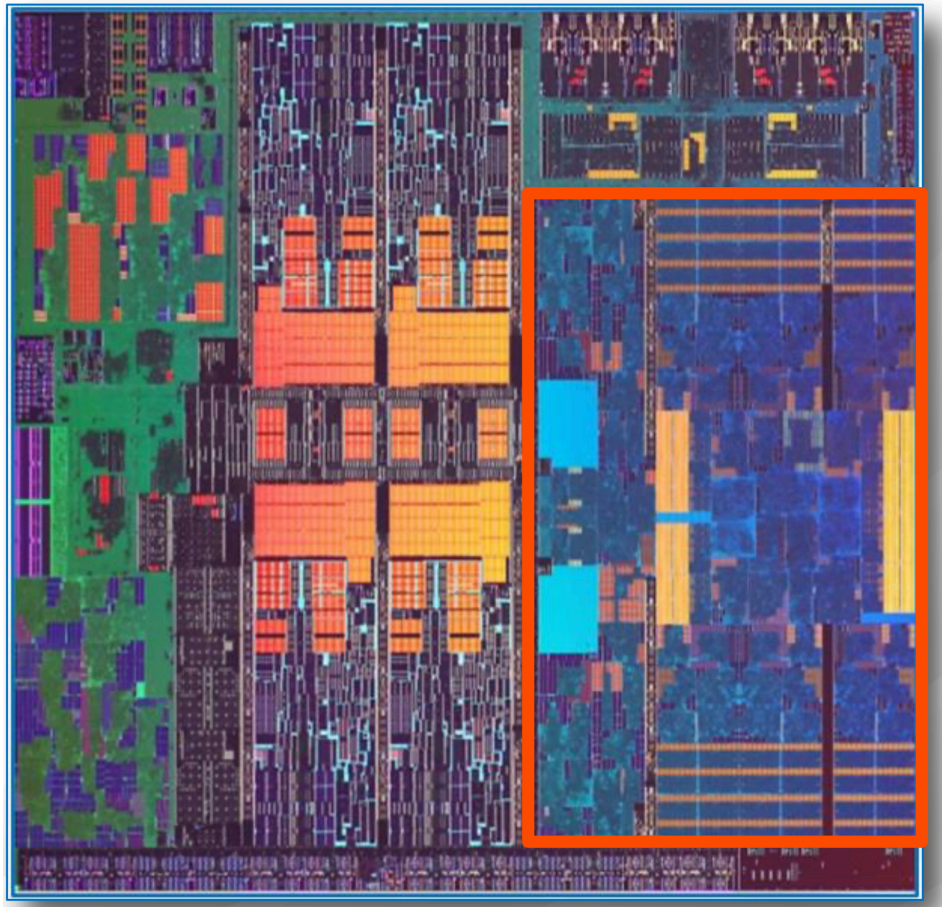
One Architecture and 4 Micro Architectures



Current Intel GPUs

- X^e LP
  - Platforms: Tiger Lake, DG1, SG1
  - Integrated & discrete
- Gen 11
  - Platforms: Ice Lake
  - Integrated
- Gen 9
  - Platforms: Skylake
  - Integrated
  - Double precision peak performance: 100-300 GF

- All have relatively low FP64 performance by design due to power and space limits

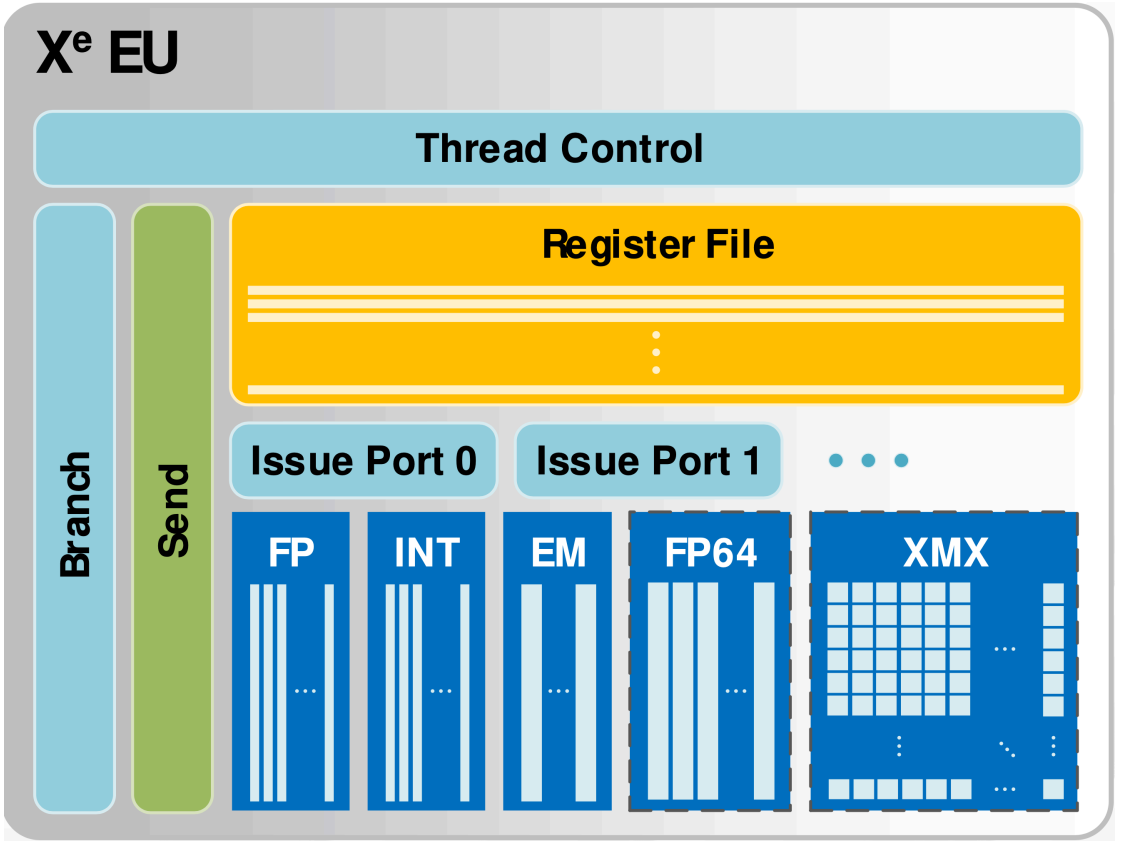


Tiger Lake SoC with X^e_{LP} GPU

XE Execution Unit

The EU executes instructions:

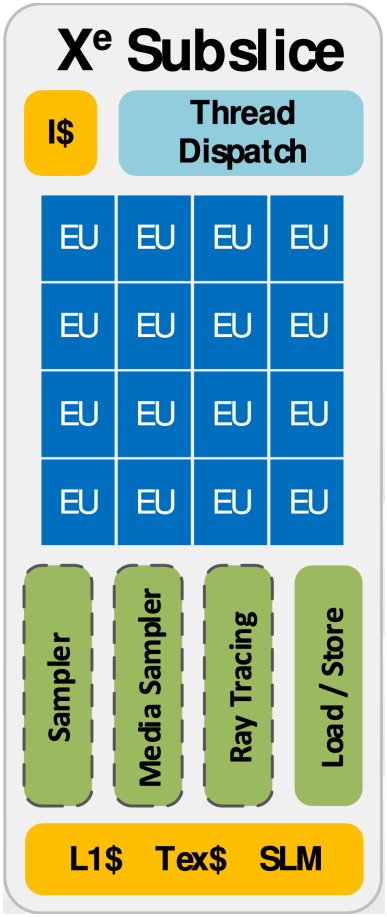
- Register file
- Multiple issue ports
- Vector pipelines:
  - Floating Point
  - Integer
  - Extended Math
  - FP64 (optional)
  - Matrix Extension (XMX) (optional)
- Thread control
- Branch
- Send (memory)



XE Subslice

A sub-slice contains:

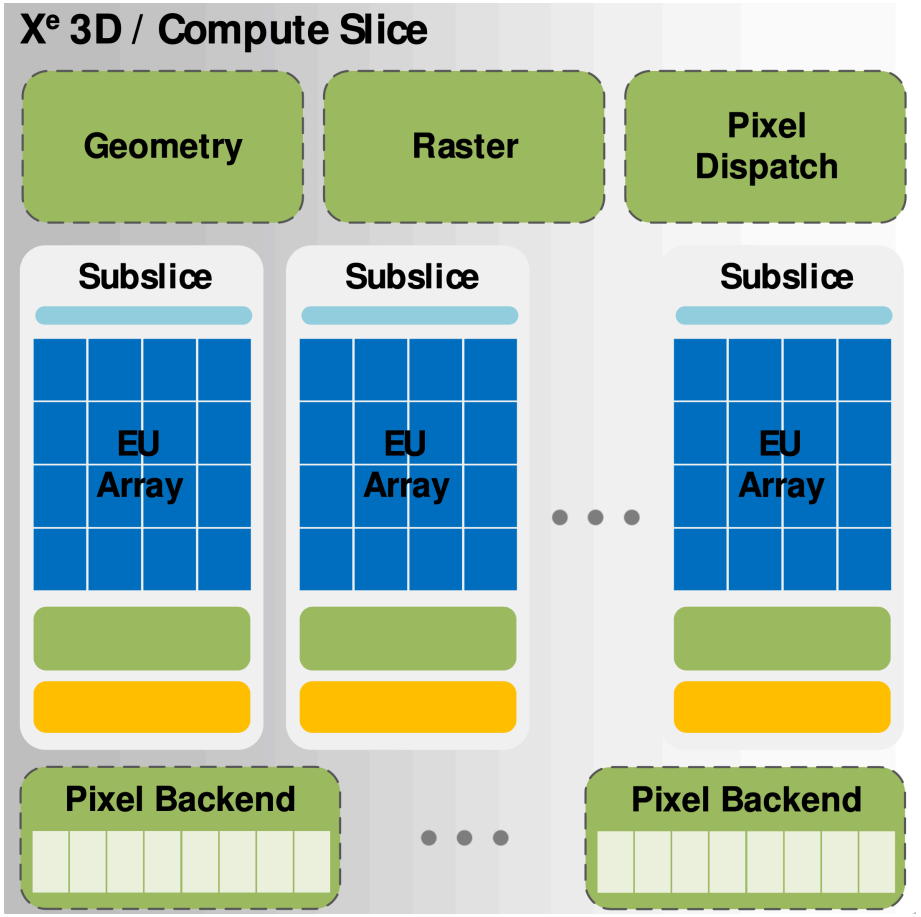
- 16 EUs
- Thread dispatch
- Instruction cache
- L1, texture cache, and share local memory
- Load/Store
- Fixed Function (optional)
  - 3D sampler
  - Media Sampler
  - Ray Tracing



Xe 3D/Compute Slice

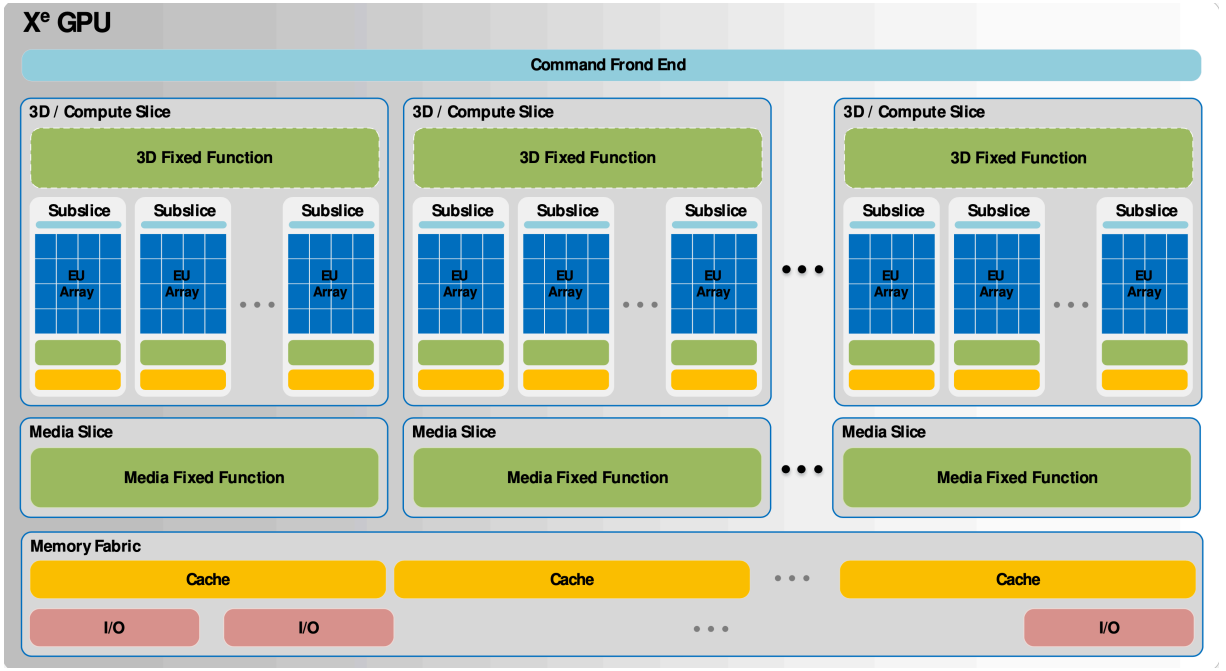
A slice contains:

- Variable number of subslices
- 3D Fixed Function (optional)
  - Geometry
  - Raster



High Level Xe Architecture

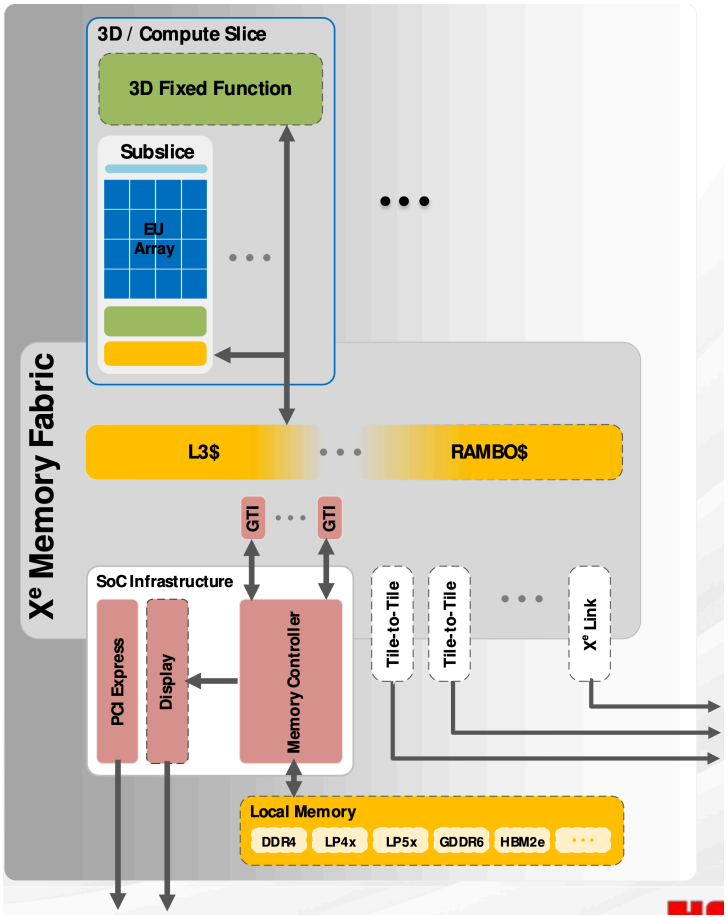
- X^e GPU is composed of:
  - 3D/Compute Slice
  - Media Slice
  - Memory Fabric / Cache



XE Memory Fabric

Coherent Scalable Interconnect Fabric

- L3 + Rambo Cache (optional)
- SoC infrastructure
  - PCIe
  - Display (optional)
  - Memory Controller
    - Local Memory (optional)



Cray Slingshot Network

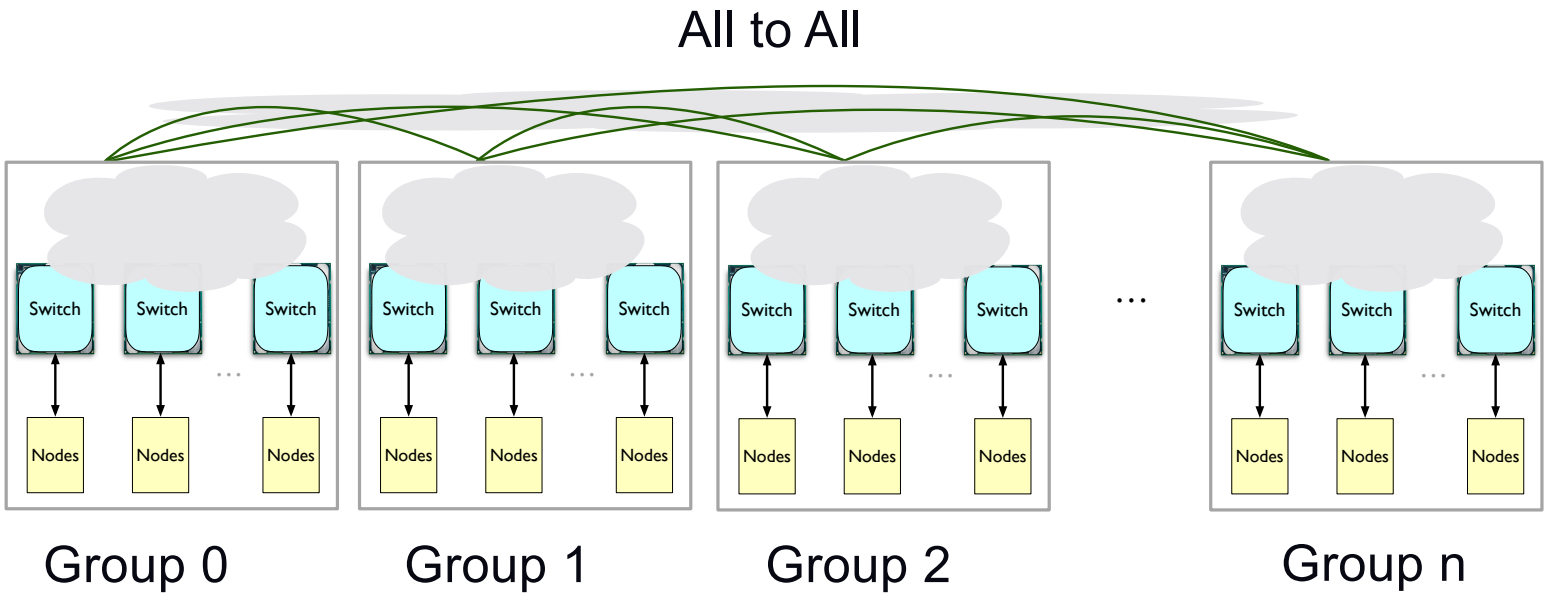
- Slingshot is next generation scalable interconnect by Cray
  - 8th major generation
- Builds on Cray's expertise in high performance network following
  - Gemini (Titan, Blue Waters)
  - Aries (Theta, Cori)
    - 5 hop dragonfly topology
- Slingshot introduces:
  - Congestion management
  - Traffic classes
  - 3 hop dragonfly

Cray Slingshot

https://www.cray.com/products/computing/slingshot
https://www.cray.com/resources/slingshot-interconnect-for-exascale-era

Dragonfly Topology

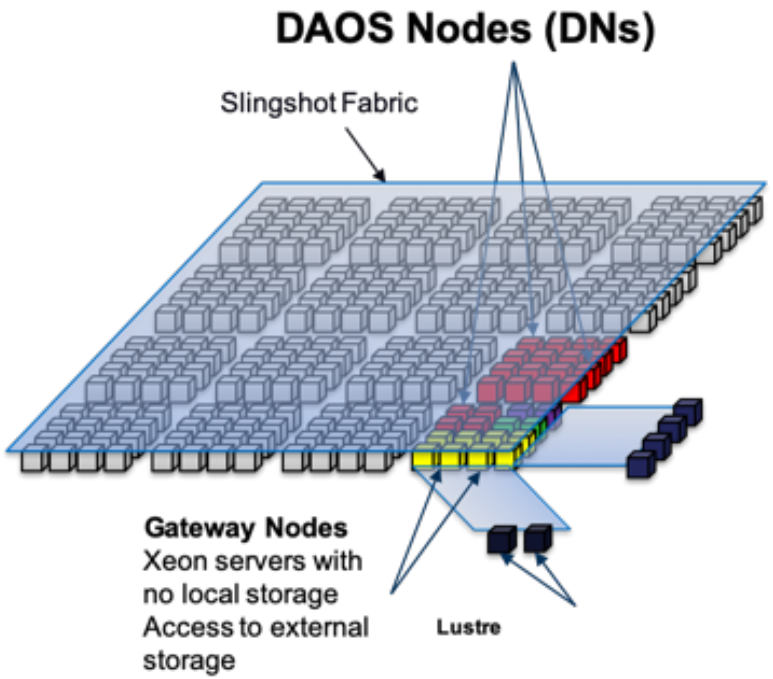
- Two layer all-to-all topology
- Nodes are organized into a number of groups
- All-to-all connectivity between nodes within the groups
- Groups are connected together in an all-to-all fashion at the group level



https://www.cray.com/sites/default/files/resources/CrayXCNetwork.pdf

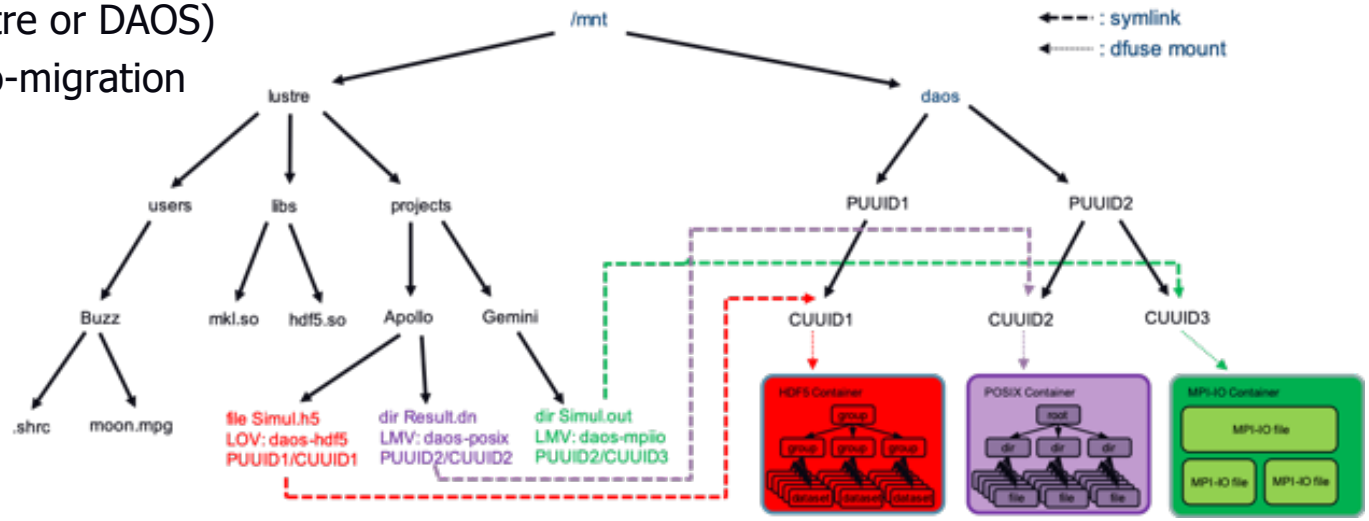
Distributed Asynchronous Object Store (DAOS)

- Primary storage system for Aurora
- High performance and capacity:
  - ≥ 230 PB capacity
  - ≥ 25 TB/s
- Persistent storage, not a burst buffer
- Provides compatibility with existing I/O models such as POSIX, MPI-IO and HDF5
- Open source storage solution
- Provides a flexible storage API that enables new I/O paradigms



DAOS and Lustre

- Aurora will provide both DOAS and Lustre file systems
- User see single storage namespace which is in Lustre
  - Links point to DAOS containers within the /project directory
  - DAOS aware software interpret these links and access the DAOS containers
- Data resides in a single place (Lustre or DAOS)
  - Explicit data movement, no auto-migration
- Suggested storage locations
  - Source and binaries in Lustre
  - Bulk data in DAOS



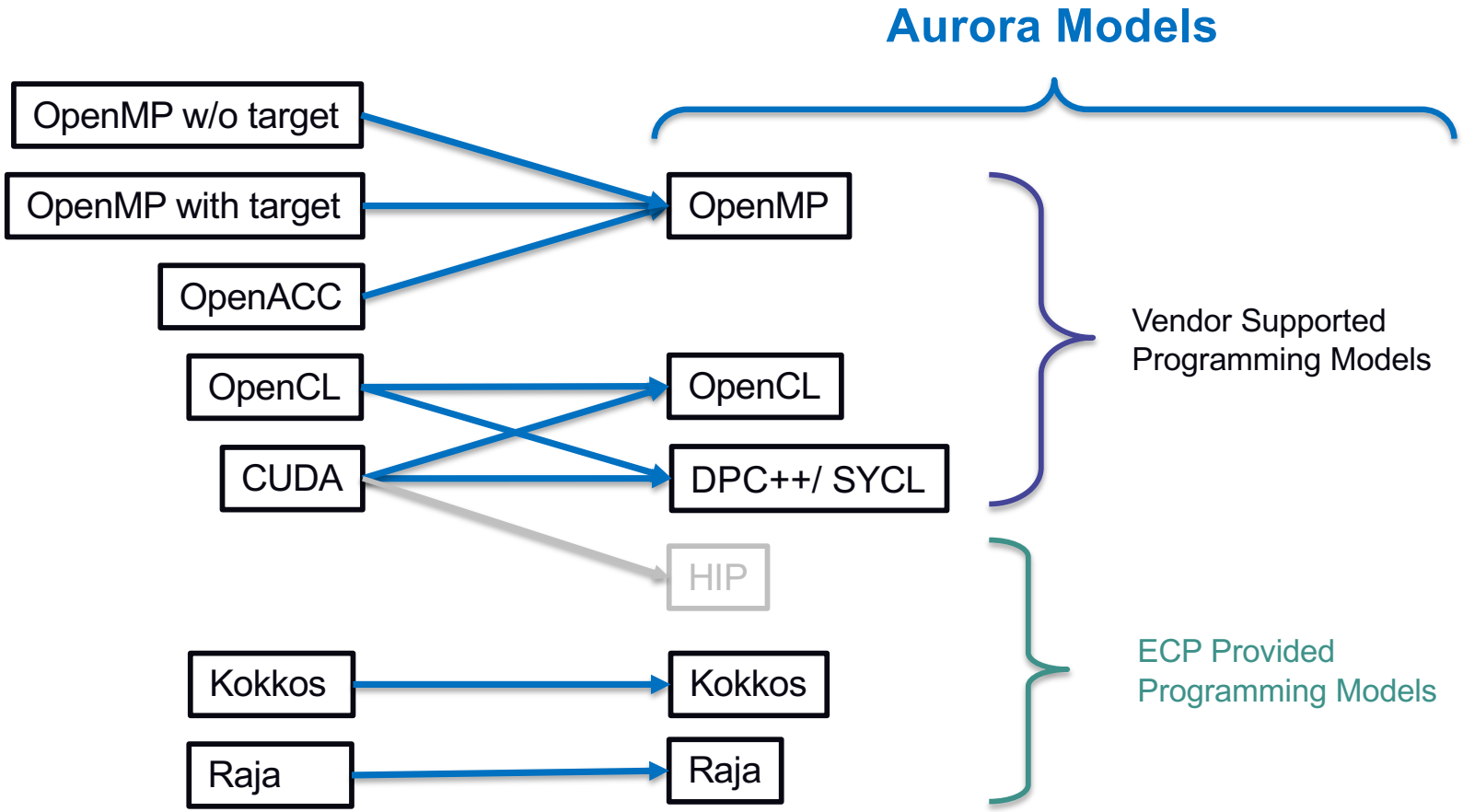
Programming Models for Exascale Systems

- Applications will be using a variety of programming models for Exascale:
  - CUDA
  - OpenCL
  - HIP
  - OpenACC
  - OpenMP
  - DPC++/SYCL
  - Kokkos
  - Raja
- Not all systems will support all models

Programming Models For Aurora

- Aurora applications may use:
  - ~~CUDA~~
  - OpenCL
  - HIP
  - ~~OpenACC~~
  - OpenMP
  - DPC++/SYCL
  - Kokkos
  - Raja

Possible Paths to Aurora



oneAPI

- Industry specification from Intel (https://www.oneapi.com/spec/)
  - Language and libraries to target programming across diverse architectures (DPC++, APIs, low level interface)
- Intel oneAPI products and toolkits (https://software.intel.com/ONEAPI)
  - Implementations of the oneAPI specification and analysis and debug tools to help programming

Aurora Software Stack

- Languages:
  - Fortran (with OpenMP 5)
  - C/C++ (with OpenMP 5)
  - DPC++
  - Python
- Libraries:
  - oneAPI MKL (oneMKL)
  - oneAPI Deep Neural Network Library (one DNN)
  - oneAPI Data Analytics Library (oneDAL)
  - MPI
- Tools:
  - Intel Advisor
  - Intel Vtune
  - Intel Inspector

Aurora Testbeds

- Intel DevCloud
  - Provides free access to GPU hardware and oneAPI software
  - https://devcloud.intel.com/oneapi/get-started/
- Local Setup
  - Download Intel oneAPI public beta
    - https://software.intel.com/content/www/us/en/develop/tools/oneapi.html
  - Run on Intel CPU with integrated graphics
- Argonne JLSE testbeds for Aurora
  - 20 Nodes of Intel Xeons with Gen9 Iris Pro integrated GPU
  - DG1 nodes
  - Intel's Aurora oneAPI SDK [NDA required]

Argonne Joint Laboratory for System Evaluation


Questions?
Argonne
NATIONAL LABORATORY